POCKET COMPUTER NEWSLETTER

© Copyright 1983 SCELBI Publications

Issue 24 — April

THE RADIO SHACK TRS-80 MODEL 100 PORTABLE COMPUTER

Radio Shack, a division of Tandy, may have scored a considerable marketing scoop with the introduction of their new portable TRS-80 Model 100.

In a break from traditional marketing methods, whereby products are announced months before they become readily available, the firm kept details of the new portable under tight wraps. Meanwhile, in a carefully orchestrated fashion, the company arranged to get substantial supplies of the new product into its outlets around the country. On March 25, 1983, the company unleashed its tongue via a newspaper media blitz and announced that the new product was in stock at local Radio Shack outlets.

The slightly under four pounds (3 lbs., 13-1/2 oz.) portable measures 11-5/8 by 8-1/4 by 2 inches, covering an area approximately equal to a standard-sized piece of writing paper. Its most prominent feature is a large 240 by 64 pixel liquid-crystal display that measures close to 2-1/8 by 7-1/2 inches. This display can present text and special graphic characters in 8 rows by 40 columns. It can also be operated in a pure pixel-by-pixel graphics mode to create pictures, charts and graphs.

The area that remains beneath the display is occupied by a full-sized touch-typing keyboard. The five-row keyboard is further complimented by 16 special function keys directly beneath the display. Additionally, a subset of the regular keyboard can be designated to serve as a numeric input keypad.

Perhaps the most surprising feature of the Model 100 is a complete built-in modem and comprehensive terminal software which permits automated telephone number lookup and dialing, user-defined log-on procedures, file up-loading and down-loading, and other aids to facilitate its use as an intelligent remote terminal.

In addition to the communications software, the unit includes a BASIC operating system, a text editor complete with word-wrap, an address-maintenance program and an appointments scheduler. The unit is supplied with a real-time clock that provides the year, month, day, and time and is accessible by software. A sophisticated memory management system permits multiple user files to reside in memory simultaneously. These user files may be accessed by the unit's operating system or by other user-created programs.

The Model 100 is pleasingly designed to facilitate its use by novices. Whenever power is turned on the display is filled with a directory giving the date and time, the names of up to 24 programs or data files currently residing in memory, and the amount of memory that is still available for general use. To access a program or data file, the user simply positions a cursor over a directory entry and presses the ENTER key. Once a file has been selected, legends at the bottom of the display, associated with special function keys directly beneath the LCD, offer options to the user in menu fashion. (The menu offerings can be toggled on and off using a function key. Thus, experienced users can free up a line of the display by not using the prompts.)

For instance, when the user initially calls up the text editing program, the system prompts for a new file name. Once given, a press of the "label" key offers choices of saving or loading a new file, finding a point in the text, selecting a portion of the text for manipulations, copying or cutting text or exiting to the main directory. In an apparent effort to make computer operations more consistent with every day experience, commands like "cut" and "paste" are used instead of "delete" or "move".

The new Radio Shack entry has a powerful BASIC interpreter, perhaps the most powerful amongst portables at this point. Tailored by Microsoft, it contains such keywords as: ATN, BEEP, CALL, CDBL, CHR$, CINT, CLEAR, CLOAD, CLOAD?, CLOADM, CLOSE, CLS, COM ON/OFF/STOP, CONT, COS, CSAVE, CSAVEM, CSNG, CSRLIN, DATA, DATE$, DAY$, DEFDBL, DEFINT, DEFSNG, DEFSTR, DIM, EDIT, END, EOF, ERL, ERR, ERROR, EXP, FILES, FIX, FOR. . .NEXT, FRE, GOSUB, GOTO, HIMEM, IF. . .THEN. . . ELSE, INKEY$, INP, INPUT, INPUT #, INPUT$, INSTR, INT, IPL, KEY, KEY LIST, KEY ON/OFF/STOP, KILL, LCOPY, LEFT$, LEN, LET, LINE, LIST, LLIST, LINE INPUT #, LINE INPUT, LOAD, LOADM, LOG, LPOS, LPRINT, LPRINT USING, MAXFILES, MAXRAM, MDM ON/OFF/STOP, MENU, MERGE, MID$, MOTOR, NAME. . .AS, NEW, ON COM GOSUB, ON ERROR GOTO, ON KEY GOSUB, ON MDM GOSUB, ON TIME$ GOSUB, ON. . .GOTO, ON. . . GOSUB, OPEN, OUT, PEEK, POKE, POS, POWER, POWER CONT, POWER OFF, PRESET, PRINT, PRINT #, PRINT USING, PRINT # USING, PSET, READ, REM, RESTORE, RESUME, RIGHT$, RND, RUN, RUNM, SAVE, SAVEM, SCREEN, SGN, SIN, SOUND, SOUND ON/OFF, SPACE$, SQR, STOP, STR$, STRING$, TAB, TAN, TIME$, TIME$ ON/OFF/STOP, VAL, and VARPTR.

Many of these extended BASIC keywords are used to control the built-in communications and input/output capabilities. In addition to the modem, the Model 100 includes interfaces and connectors to: store and retrieve programs and data on an audio cassette recorder, drive an external printer (parallel "Centronics" interface), and connect to serial devices via an RS-232C (25-pin) connector.

The Model 100 is currently being supplied in two standard initial configurations: equipped with 8K of user RAM or 24K of user RAM. Both versions are said to be internally expandable to a maximum of 32K of RAM. The 8K unit is retailing for $799.00. The 24K unit is retailing for $999.00.

The unit is powered by four AA cells, which are said to power an 8K unit for approximately 30 days or a 32K unit for approximately 8 days. A low power light-emitting diode turns on when batteries need replacement. The unit may also be powered by an optional a.c. power pack.

The machine is supplied with an impressive 224-page (8-1/2 by 11 inch) operating manual. The manual is well organized, illustrative yet concise. A soft carrying case is also provided with the computer.

Surprisingly, batteries are not supplied with the unit. You will have to spring for 4 AA cells or an a.c. adaptor to power up the machine. Additionally, while the direct-connect modem and terminal software is supplied as part of the computer, the connector that goes to a telephone is an "optional accessory". Radio Shack is charging $19.95 for this special connector, but includes manuals and introductory time on the Compuserve and Dow Jones information networks as part of the package. You will also need to shell out $5.95 for a connector that mates to a cassette recorder if you want to store/retrieve information on magnetic tape.

The Model 100 appears to be an impressive entry to the compact portable computer market. Take a look at it.

PC-1250/51 PROGRAMMABLE IN MACHINE LANGUAGE?

It does appear that the new PC-1250 (and the PC-1251) may be programmable in machine language! If so, this may help boost the low-priced unit to the stratosphere of sale levels.

Preliminary explorations have revealed that the unit does respond to PEEK and POKE directives, even though these capabilities are not indicated in instructions accompanying the unit. The PEEK directive takes the format: PEEK X where X is an address in the range 0 - 65,535. The POKE directive has the form: POKE X,Y where X represents an address value and Y indicates a value in the range 0 - 225 which is to be stored at the stated address. An alternate form of this directive is: POKE X,Y, . , . , . , where successive values are stored in consecutive locations in memory.

Memory Mapping

If you use PEEK to start examining areas in memory, you will find some surprises. For instance, peeking into an address that does not contain any RAM or ROM results in the "page" of the address used being echoed. Thus, PEEK 0 will return the value 00. (Oh, yes, you can use decimal or hexadecimal notation. Preceding a value by an ampersand (&) causes it to be interpreted as a hexadecimal value, just as is the case on a PC-1500.)

There does not appear to be any RAM or ROM in the address range 0 - 16383 (0 - &3FFF). Addresses 16384 - 32767 (&4000 - &7FFF) appear to contain ROM directives. Addresses 32768 - 34815 (&8000 - &87FF) apparently access user RAM. However, the same memory elements can also be reached using address &8800 - &8FFF or &9000 - &97FF, etc.: in other words, at 2K intervals. (Thus, it appears that RAM does not use complete address decoding.) This continues until address 49152 (&C000) is reached. In a PC-1251, additional RAM appears at 49152 (&C000) and this block is duplicated beginning at 53248 (&D000).

Theoretically, at least according to the advertising literature, there should be another 8K of ROM somewhere in the unit. (The literature clearly says the PC has 24K of ROM.) At this time, the search for that ROM is still in progress. It is quite likely that this block of coding handles the CE-125 device. Unlike the PC-1500, the PC-1250 properly tokenizes commands such as LPRINT whether or not it is connected to the printer unit. Provisions must have been made to interpret such statements within the basic PC.

ROM Coding

Peeking at ROM (beginning at address &C000) reveals quite a bit of information. First of all, it quickly becomes evident (to those that may have delved into the PC-1500) that the CPU chip is *not* the same as that used in the PC-1500! The machine language instruction set is different. It is a little early to tell just how much different, in terms of architecture and so forth, but at this point it appears that it certainly does not use the same machine codes!

Furthermore, scanning the ROM for familiar ASCII characters (to locate token tables, for instance) won't do much good. You see, the PC-1250 (like its predecessor the PC-1211 and Radio Shack equivalent PC-1) doesn't use ASCII codes at the machine level! The letter A is represented by &51 (instead of &41), digits are in the range &40 - &49 (instead of &30 - &39) and symbols and punctuation marks are different, too (space = &11 instead of &20). You need to remember this if you examine BASIC source code stored in user's RAM. Can you imagine the gyrations the BASIC interpreter has to go through when dealing with strings? Remember, the specifications say that string functions (such as ASC) return true ASCII values! Whew.

In any event, at address &412A in ROM you can pick up what appears to be the main token table. The first entry there is: B5 51 62 55 51 54 DC 51 26. This can be decoded as: a classifier byte (B5) with the 5 part indicating that the keyword has 5 characters. The next five bytes, 51 62 55 51 54, represent the letters A R E A D, which is simply the AREAD keyword. Note the non-ASCII representation of the characters. Next comes the token value for that keyword (DC). Finally, the next two bytes probably represent the address in ROM (51 26) where the routine to process the statement begins. The table continues in this fashion and contains a number of functions as well as commands and statements. If you are inclined toward working on cracking the machine codes used in this PC, this is a good place to start getting a feel for the layout of the ROM. Not surprisingly, the tokens for all of the statements that appeared in the original PC-1211 are the same on the PC-1250. Of course, new tokens appear to represent the upgraded capabilities. (The fact that this would virtually have to be the case is hinted at in that the PC-1250 can interpret source code stored on tape by a PC-1211. This source code is in tokenized form!)

Do Your Own Dump(s)

If you want to do your own snooping around, you can begin using the routine shown in the accompanying listing. It is a modification of an earlier routine used to print out the contents of memory in a PC-1500. (Refer to *PCN* Issue 15, page 3.) In order to conserve paper, locations are dumped in blocks of 64. Even so, a complete roll of thermal paper will only display a few kilobytes of memory. Choose your targets carefully! Happy hunting!

Program *Memory Dump for PC-1250*

```
10:"D" CLEAR : INPUT "S
   TART ADDR? ";A
12:INPUT "ENDING ADDR?
   ";F
15:DIM B$(1)*24
20:B$(0)="0123456789ABC
   DEF"
25:W= INT (A/4096)
26:X= INT ((A-(W*4096))
   /256)
27:Y= INT ((A-(W*4096)-
   (X*256))/16)
28:Z= INT ((A-(W*4096)-
   (X*256)-(Y*16)))
29:LPRINT MID$ (B$(0),W
   +1,1); MID$ (B$(0),X
   +1,1); MID$ (B$(0),Y
   +1,1); MID$ (B$(0),Z
   +1,1);"  "
30:FOR B=0 TO 7:C=(
   PEEK (A+B)) AND (&F0
   ):D=1+ INT (C/16):E=
   ( PEEK (A+B)) AND (&
   0F)
40:B$(1)=B$(1)+ MID$ (B
   $(0),D,1)+ MID$ (B$(
   0),E+1,1)+" ": NEXT
   B
45:LPRINT B$(1):B$(1)="
   ": IF INT ((A+8)/64)
   > INT (A/64) LET G=2
   5: GOTO 55
50:G=30
55:IF A+7<F LET A=A+8:
   GOTO G
```

FILE MAINTENANCE PROGRAM

The file maintenance program allows the setup of a filing system to a user's specific needs with regard to record size, number of fields, and field length. The complete file is memory resident for purposes of access. The record size is dependent upon the number of fields and their length. Once a file has been created, records may be added, changed or retrieved and the file may be saved on cassette. The program was developed on the Radio Shack PC-2, and I suspect it will run on the Sharp PC-1500. Using an 8K memory module, the following conditions are possible:

Record Length	80 Bytes Max
Number of Records	80 Max (if record length is 80)
Fields per Record	20 Max
File Name	16 Bytes Max (fixed)
Field Name	16 Bytes Max (fixed)
Field Length	80 Bytes Max
Field Search	5 Fields Max

To run the program, execute a run command and a continuous menu will be displayed. To select a menu item, a single key stroke of the appropriate letter or number is all that is necessary. The program prompts the user and validates all inputs. An invalid input will cause a return to the prompt. Capacity overflows are indicated by error messages. The execution of the menu subroutine executes a clear command; therefore, using the Break key and restarting the program will erase any previous files and data in memory. Upon the completion of a menu function, the menu will be displayed. The following menu functions are available:

S – Setup new file
O – Output file
I – Input file
1 – Print file parameters
A – Add a record
P – Print a record
F – Find record(s)
C – Change a record

A delete function was not incorporated in order to conserve program space. If a record is created and is no longer needed, label one of its fields deleted. When a new record is desired, use the Find function to locate the record and the Change function to edit its contents to the new data.

If peripheral programs are to be written on the data base, the following file information will be helpful:

F$	=	File name
N$ (array)	=	Field names
R$ (array)	=	Records
L (array)	=	Length of fields
F	=	Number of fields in record
R	=	Number of records in file
X	=	Length of record
U	=	Number of records used

This program submitted by: *Stephen Tomback, 19 Maplewood Way, Pleasantville, NY 10570.*

Program *File Maintenance*

```
 1:REM File maint
   enance
 2:REM FM 10/11/8
   2  12/04/82
10:REM MENU
15:CLEAR :LOCK :
   DIM U$(0)*80
20:CLS :WAIT 0:E=
   8:K=0:RESTORE
   :FOR I=1TO E:
   READ U$(0):
   PRINT U$(0)
25:FOR J=1TO 60:U
   $=INKEY$ :IF U
   $<>""LET J=60
30:NEXT J
35:IF U$=""THEN 5
   5
40:FOR J=1TO E:IF
   MID$ ("SOI1APF
   C", J, 1)=U$LET
   K=J:J=E
45:NEXT J
50:IF KCLS :ON K
   GOSUB 100, 400,
   500, 600, 700, 85
   0, 900, 1000:
   GOTO 20
55:NEXT I:GOTO 20
60:DATA "S - SETU
   P NEW FILE", "O
    - OUTPUT FILE
   ", "I - INPUT F
   ILE"
65:DATA "1 - PRIN
   T FILE PARAMET
   ERS", "A - ADD
   A RECORD", "P -
    PRINT A RECOR
   D"
70:DATA "F - FIND
    RECORD(S)", "C
   - CHANGE A RE
   CORD"
100:REM NEW FILE
105:CLEAR :INPUT "
   NEW FILE NAME:
    ";F$
110:IF LEN F$<1
   THEN 105
115:INPUT "HOW MAN
   Y RECORDS: ";R
120:IF R<1OR R>255
   THEN 115
125:INPUT "FIELDS
   PER RECORD: ";
   F
130:IF F<1OR F>20
   THEN 125
135:DIM N$(F-1):
   DIM L(F-1)
140:FOR I=1TO F
145:CLS :PRINT "FI
   ELD";I;" NAME:
   ":CURSOR 14:
   INPUT "";N$(I-
   1)
150:IF LEN N$(I-1)
   <1GOTO 145
155:CLS :INPUT "HO
   W MANY BYTES:
   ";L(I-1)
160:IF L(I-1)=0
   THEN 155
165:IF X+L(I-1)<81
   LET X=X+L(I-1)
   :GOTO 175
170:PRINT 80-X;" B
   YTES LEFT":
   BEEP 15:GOTO 1
   55
175:NEXT I
180:IF STATUS 3-
   STATUS 2-R*X-1
   81<1THEN 190
185:DIM R$(R-1)*X:
   R$(0)="":DIM I
   $(0)*80:DIM U$
   (0)*80:RETURN
190:PRINT "NOT ENO
   UGH MEMORY":
   BEEP 15:GOTO 1
   05
400:REM OUTPUT
405:IF LEN F$=0
   GOSUB 2000:
   RETURN
410:INPUT "TAPE RE
   ADY? Y to outp
   ut ";I$
415:IF I$<>"Y"THEN
   410
```

Program *File Maintenance (conclusion)*

```
420:PRINT #F$;F,R,
    X,U
425:I$=F$+"*"
430:PRINT #I$;N$(*
    ),R$(*),L(*):
    RETURN
500:REM INPUT
505:CLEAR :INPUT "
    INPUT FILE NAM
    E: ";F$
510:INPUT #F$;F,R,
    X,U
515:DIM N$(F-1):
    DIM R$(R-1)*X:
    DIM L(F-1)
520:I$=F$+"*"
525:INPUT #I$;N$(*
    ),R$(*),L(*)
530:DIM I$(0)*X:
    DIM U$(0)*X:
    RETURN
600:REM PARAMETERS
605:IF LEN F$=0
    GOSUB 2000:
    RETURN
610:TEXT :CSIZE 1:
    LF 5
615:LPRINT "FILENA
    ME : ";F$
620:LPRINT "NO OF
    RECORDS :";R
625:LPRINT "RECORD
    S USED :";U
630:LPRINT "BYTES
    PER RECORD :";
    X
635:LPRINT "MEMORY
     LEFT :";
    STATUS 3-
    STATUS 2:LF 1
640:FOR I=1TO F:
    LPRINT "FIELD"
    ;I;" ";N$(I-1)
    ;TAB 28;USING
    "###";"BYTES";
    L(I-1):USING
645:NEXT I:LF 6:
    RETURN
700:REM ADD
705:IF LEN F$=0
    GOSUB 2000:
    RETURN
710:FOR I=0TO R-1:
    REM FIND 1ST A
    VAILABLE RECOR
    D
715:IF LEN R$(I)=0
    THEN 725
720:NEXT I:PRINT "
    NO MORE RECORD
    S!";BEEP 15:
    RETURN
725:FOR J=0TO F-1
730:CLS :I$(0)="";
    PRINT N$(J);":
     ";CURSOR LEN
    N$(J)+2:INPUT
    "";I$(0)
735:IF LEN I$(0)>L
    (J)THEN 760
740:IF LEN I$(0)=L
    (J)THEN 750
745:I$(0)=I$(0)+"
    ";GOTO 740
750:R$(I)=R$(I)+I$
    (0):I$(0)=""
755:NEXT J:U=U+1:
    CLS :PRINT "AD
    D RECORD";U:
    BEEP 15:RETURN
760:GOSUB 3000:
    GOTO 730
800:REM OUT TO PRI
    NTER
805:TEXT :CSIZE 1:
    LF 1:J=0
810:LPRINT "RECORD
    :";N:LF 1
815:FOR I=0TO F-1
820:IF I=0LET J=1
825:LPRINT N$(I);"
    : ";MID$ (R$(N
    -1),J,L(I)):J=
    J+L(I)
830:NEXT I:LF 1:
    RETURN
850:REM PRINT
855:IF U=0GOSUB 20
    00:RETURN
860:N=0:INPUT "REC
    ORD NUMBER: ";
    N
865:IF N<1OR N>U
    THEN 860
870:GOSUB 800:LF 5
    :RETURN
900:REM  FIND
905:IF U=0GOSUB 20
    00:RETURN
910:CLS :Q=0:INPUT
    "HOW MANY FIEL
    D SEARCH? ";Q
915:IF Q<1OR Q>5OR
    Q>FTHEN 910
920:RESTORE 990:
    FOR I=1TO Q:
    READ W$:Z$=
    STR$ I
925:CLS :@(I)=0:
    CURSOR :WAIT 0
    :PRINT Z$;W$:
    CURSOR 4:INPUT
    "FIELD NUMBER:
     ";@(I)
930:IF @(I)<1OR @(
    I)>FTHEN 925
935:CLS :@$(I)="";
    CURSOR 0:PRINT
    "SEARCH";@(I);
    " FOR: ";
    CURSOR 14:
    INPUT "";@$(I)
940:IF LEN @$(I)>L
    (@(I)-1)GOSUB
    3000:GOTO 935
945:IF LEN @$(I)<1
    THEN 935
950:NEXT I
955:FOR P=0TO U-1:
    FOR I=1TO Q
960:CLS :PRINT "RE
    CORD";P+1:BEEP
    1
965:S=@(I):GOSUB 4
    000
970:T=LEN @$(I)
975:IF MID$ (R$(P)
    ,Z,T)<>@$(I)
    NEXT P:RETURN
980:NEXT I
985:N=P+1:LF 2:
    GOSUB 800:NEXT
    P:RETURN
990:DATA "st","nd"
    ,"rd","th","th
    "
1000:REM CHANGE
1005:IF U=0GOSUB
     2000:RETURN
1010:N=0:INPUT "R
     ECORD NUMBER
     : ";N
1015:IF N<1OR N>U
     THEN 1010
1020:P=0:INPUT "F
     IELD NUMBER:
      ";P
1025:IF P<1OR P>F
     THEN 1020
1030:I$(0)="";
     INPUT "CHANG
     E TO: ";I$(0
     )
1035:IF LEN I$(0)
     >L(P-1)THEN
     1050
1040:IF LEN I$(0)
     =L(P-1)THEN
     1055
1045:I$(0)=I$(0)+
     " ":GOTO 104
     0
1050:GOSUB 3000:
     GOTO 1030
1055:IF P=FTHEN 1
     080
1060:S=P+1:GOSUB
     4000:T=X-Z+1
1065:U$(0)=MID$ (
     R$(N-1),Z,T)
1070:I$(0)=I$(0)+
     U$(0)
1075:IF P=1LET R$
     (N-1)=I$(0):
     RETURN
1080:S=P:GOSUB 40
     00:R$(N-1)=
     MID$ (R$(N-1
     ),1,Z-1)+I$(
     0):RETURN
2000:PRINT "MEMOR
     Y EMPTY!";
     BEEP 15:
     RETURN
3000:CLS :PRINT "
     FIELD LENGTH
      EXCEEDED!":
     BEEP 15:
     RETURN
3336:+6701
4000:REM START PO
     S OF FIELD I
     N R$
4001:REM  ENTER/F
     IELD # IN R$
4002:REM  EXIT/PO
     S IN Z
4005:Z=1:IF S=1
     RETURN
4010:FOR K=1TO S-
     1:Z=Z+L(K-1)
     :NEXT K:
     RETURN
STATUS 1      3256
```

THE CE-158 RS-232C AND PARALLEL INTERFACE

After quite a few hours of staring at the manual, it became apparent that you needed more than just the CE-158 in order to make anything exciting occur! Things began to happen when I spotted my idle Epson MX-80 line printer. But wait! The connector on it didn't look anything like the parallel output on the CE-158. No kidding! They are not the same. One is a 25-pin EIA connector, the other is a 36-pin "alligator" job. Whew. That seems ridiculous. What do you want to bet you have to buy a cable that has the two different mating connectors on each end just to make it work? Would you believe Sharp is retailing it for $75.00? Believe it.

Of course, if you do not want to shell out that kind of money for a connecting cable, take a look at the accompanying table. If you can find an ordinary female 25-pin RS-232C connector and a male 36-pin alligator ("Centronics") connector, then you can wire up your own cable. Of course, it is still going to cost you a few bucks because those connectors and multi-wire cable are not cheap. But, seventy-five bucks?

Table *Cable Wiring from CE-158 Parallel Port to Centronics Connector*

EIA	Centronics
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	11
11	31
12	nc
13	nc
14	19
15	20
16	21
17	22
18	23
19	24
20	25
21	26
22	27
23	29
24	30
25	16

The cable, though, is just about half of the trick. The other half is simply using the statement: OPN "LPRT". This short statement is all it takes to redirect data from the CE-150 printer to your external unit! Once issued, a command such as LLIST will cause programs to be listed on your full width printer. Execution of LPRINT statements similarly are routed through the parallel port to your desktop clunker.

There are a few other special directives to ease the path. The FEED statement causes your external printer to linefeed. If you want 8 linefeeds in a row, just use: FEED 8. Another directive, CONSOLE, is used to format the maximum lengths of lines outputted on the external device. This appears to be settable from 16 characters to 80 on my system. (Remember, it is an Epson MX-80). You use this directive once when you first start using the interface. That format will remain until you change it (or power down). There is a suffix to this directive that is supposed to allow alteration of the "end of line" character. However, the details of this specification are unclear at this point. I have been getting away with copying a statement example wherein this suffix value was 1. Thus, to set a line length of 40 characters, I just execute: CONSOLE 40,1.

While the CE-158 Instruction Manual is pretty much a disaster, the section relating to the parallel port (pages 64 - 71) should be generally decipherable by those with an electronic technician's bent. This is primarily because it consists of charts and tables. There is even a timing diagram so that designers can take a whack at the using the port for other output purposes. (It sure looks as though it should be possible for someone to design a video interface that accepted data from this port.)

The RS-232C Port

Trying to get anything to happen with the RS-232C port can take a lot of patience. It turns out, I think, that most people experience a lot of difficulty with this part of the interface because of one particular operational aspect. (That is, however, after discounting the effects of the woeful manual, which can cause enough grief all by itself!)

The CE-158 is capable of communicating through the RS-232 port using a set of special BASIC statements. Or, it can be placed in a powerful TERMINAL mode. (No, it doesn't cause the unit to "drop dead", but you might tend to wish it would while you are struggling to understand its operation using the manual supplied.) When in the terminal mode, the PC-1500 will effectively emulate a terminal with many operational features. All of the programming for accomplishing terminal operations, including a variety of options, are on ROM inside the CE-158. It is all very nice, once you get it working for you.

Where Beginners Have a Problem

Naturally, most people, when they get a new item, like to "take it for a spin". Trouble is, if you don't know what you are doing with the CE-158, you can not really tell whether its wheels are turning! The trouble centers around that pesky little RS-232C port.

An EIA RS-232C port is a two-way interface. It is designed to both transmit and receive data. It is also quite flexible. Indeed, this flexibility can cause a lot of problems for the uninitiated. Through the use of sometimes complex arrangements of signals, the interface can be adapted to a number of communication "protocols" (systems or methods of passing information).

The RS-232C interface includes a number of control signals with names such as Request To Send (RTS), Clear To Send (CTS), Data Set Ready (DSR) and Carrier Detect (CD). These signals are used to tell the interface when it is appropriate to transmit data to another device. Of the four signals just listed, one is an "output" signal from the CE-158 (in this case) which is used to tell another device (such as a modem) that the CE-158 has information to transmit. This is the RTS (Request To Send) signal. The other three signals are "inputs" to the CE-158. They must all be in the proper state (a logic one or "marking") before the unit will begin electronically transmitting information on pin 2 (TD – Transmit Data) of the interface.

Thus, before the software that controls the CE-158 will allow it to send information out of the RS-232 port, all three of the following RS-232 pins must be at a logic one: pin 5 – the Clear To Send (CTS) lead, pin 6 – the Data Set Ready (DSR) lead, and pin 8 – the Carrier Detect (CD) lead. (Remember, a logic one at an EIA RS-232C interface is defined as a distinctly positive voltage. For the CE-158 it is spec'd to be between +5 and +15 volts relative to the ground reference on pin 7 of the port.)

In a typical, full-dress communications system, those pins would be fed signals from a device, such as a modem, in such a manner that when it was time for the CE-158 to transmit, it would be able to do so. That is, the logic levels would be appropriate at those inputs.

But, what happens if someone just wants to "fool around" with their CE-158 to "see if it works"? What happens if you try to get the CE-158 to send some data "to nowhere"? In other words, what will happen if someone tries to check out the RS-232C port when nothing is connected to it?

It turns out that the CE-158 is designed so that nothing will happen under this condition! And, I would not be surprised to learn that more than one purchaser has returned their CE-158 to the factory after they mistakenly thought it was malfunctioning because they attempted to perform such a test!

Unless you fool the CE-158 into thinking that it is connected to a device in a communications system, the unit will just cause the PC-1500 to "hang up" (go into an endless loop) when you try to execute directives intended to have it transmit data!

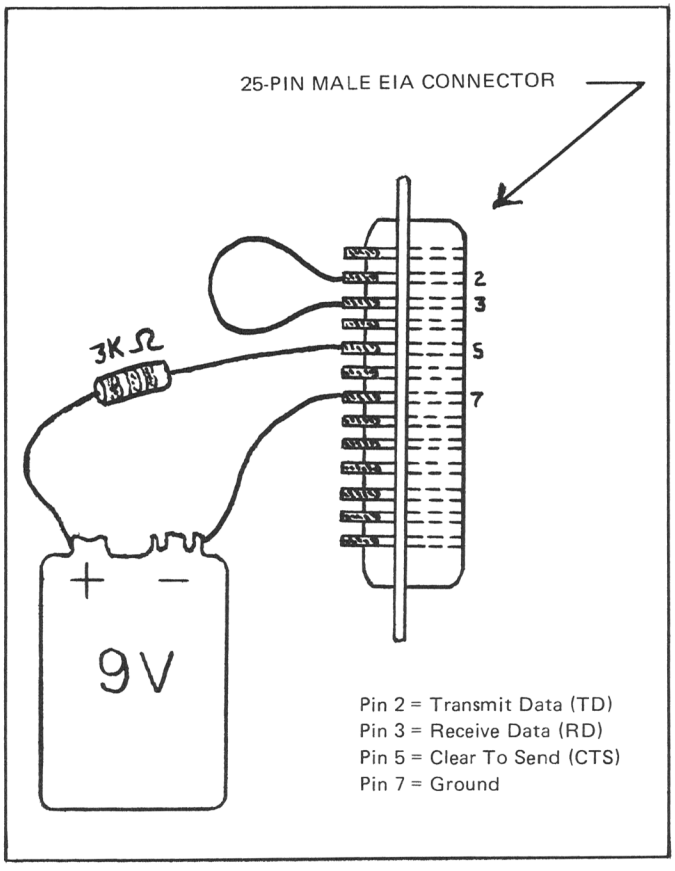
However, tricking the interface into thinking it is connected to something is not difficult, once you understand the situation. It turns out that two of the control pins on the RS-232 port effectively "float" to the logic high state when they are not connected to another device. Pins 6 and 8 (Data Set Ready and Carrier Detect) automatically assume this logic one condition when they are not connected elsewhere. However, the Clear To Send signal on pin 5 is designed to "sink" to a logic zero level when it is disconnected. That one signal will prevent the interface from transmitting data. When it is at a low state, it will cause any software that attempts to execute a data transmit directive to effectively "hang up". In short, it can stop the CE-158 interface in its tracks!

Make Your Own Test

If you are one of those people who have been pounding your head against the wall because, for instance, you could not get a short test program (such as that at the bottom of page 24 in the CE-158 Instruction Manual) to complete execution or couldn't make the TERMINAL mode operate properly when hooked to a simple modem, you might want to try the following test.

Obtain a small 9-volt radio battery, a couple of clip leads, a 3000 ohm resistor, and a male 25-pin EIA connector. Clip the minus terminal of the battery to pin 7 of the connector. Attach the 3000 ohm resistor to the plus lead of the battery. Clip the other end of the resistor to pin 5 of the connector. Plug the male connector into the RS-232C port (lower connector) on the CE-158.

Pictorial *CE-158 Test Circuitry*



Try executing a few of those directives or short programs (such as the one on page 24 of the manual) that used to get hung.

Not convinced? Do you really want to see your RS-232 port "talk to itself"? Try this: Add a jumper between pins 2 and 3 of the test connector that you have connected to the port. This will effectively "loop-back" data from pin 2 (output) to pin 3 (input). Place the CE-158 in the terminal mode by typing the TERMINAL command (or the similar DTE directive) on the keyboard of the PC-1500 (when it is in the RUN mode). When the menu:

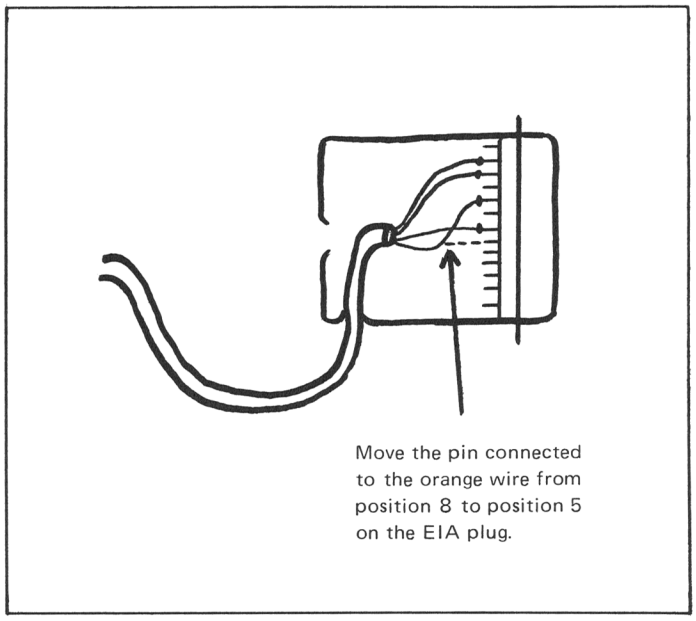
Terminal: Ent Aut Quit

comes up, select the "Ent" option. Now type some data. You should see the characters you input reflected on the LCD of the PC-1500. (If you had been fooling around with various terminal options and had placed ECHO ON, then you would see double sets of characters each time you pressed a key.) This demonstrates that what you are typing on the keyboard is being transmitted out pin 2 and is coming back in pin 3 in the test arrangement. If you want to see the effect of the artificial signal level at pin 5 (Clear To Send), remove the clip lead between the battery, the 3K resistor, and pin 5. Try typing some more data on the keyboard. What happened? Replace the clip lead to raise the signal level back to a logic one. Can you see the keyboard characters echoed again?

Hooking Up a J-CAT Modem

Of course, a real fun way to use the CE-158 is to hook it up to a communications modem so that you can talk with a big computer over the phone lines. It can be a bit incongrous these days to have a system wherein the PC itself is half the size of a modem, but that does not have to be the case any longer. A company by the name of Novation, Inc., is now producing a miniature modem that they call the J-CAT. It is a type 103 (originate/auto answer) model. It is not fancy, but it should be adequate for many PC users. The list price is $149.95 in the United States. Remember, you need this type of device (a modem) in addition to the CE-158 if you want to communicate over the phone lines using your PC.

Pictorial *Modifying the EIA-to-Modem Cable for the J-CAT Modem*



The J-CAT modem comes from the factory configured as a very simple communications modem. It is not wired up to perform fancy protocols via Request To Send, Clear To Send, Data Set Ready signals, etc. Nope, it is supplied with just four wires going to a 25-pin EIA connector: pin 2 (Transmit Data), pin 3 (Receive Data), pin 7 (Ground) and pin 8 (Carrier Detect). Alas, if you simply plug this connector from the J-CAT to the CE-158, you will find that you do not have a working

communications system.

But, by now you should know why! Yep, it is because of that pin 5 on the RS-232 port of the CE-158 which sinks to a logic low level when nothing is connected to it. Remember, the connector from the J-CAT modem does not normally use pin 5.

However, there is a quick and easy remedy to the situation. Since pin 8 (Carrier Detect) on the CE-158 port will float high if nothing is connected to it, the signal coming from the J-CAT to that pin is not really needed in this system. Instead, the Carrier Detect signal from the modem can be used to activate pin 5 (Clear To Send) on the CE-158.

The EIA connector supplied with the J-CAT has removable pins. It is a simple matter to push out the connector pin on the cable supplied with the J-CAT at pin 8 (which is connected to the wire carrying the Carrier Detect signal) and move it over so that it is at pin position 5. (Refer to the accompanying diagram.) That is all it takes to construct a viable modem system for use with networks such as The Source or Compuserve!

Communicating Using the CE-158

Once a modem such as the Novation J-CAT is successfully connected to the CE-158 and the phone lines, you are ready to tap the built-in communications capabilities of the CE-158. (It is assumed at this point that you are familiar with networks such as Compuserve and The Source and want to use such services. By the way, the J-CAT modem is currently being delivered with a package that enables you to try out The Source for one hour at no charge.)

Typing in the keyboard statement TERMINAL or DTE will automatically place the CE-158 into its terminal communications mode. This is initially indicated by a brief display of the message:

---ENTER MENU SELECTION. . .

followed by the beginning of what is a multi-part menu:

Terminal: Ent Aut Quit

You access various options by pressing the softkeys underneath the menu legends. Or, you can bring up another part of the primary menu by, for instance, pressing the line scroll (down) key. Doing so at this point, for instance, would reveal a new selection:

Setup: Aut Fnc Com

Another press of the scroll (down) key would yield:

Operate: Nrm A/P A/L

and yet another would reveal:

Protocol: XO/O Echo

At this point, using the line scroll key will cause the original menu selection to reappear. In other words, this portion of the communications menu cycles in the display (in response to pressing the line scroll keys) as though it were on a cylinder or drum.

Another set of options is accessed by pressing the SHIFT and line scroll (up) keys:

Output: Ext Trc Dsp Etx

This separate group of options is related to output or display options when in the communications mode.

To give you an idea of how various communications options may be selected, consider the output group just mentioned. If the softkey underneath the legend "Ext" was pressed, the PC display would change to query:

EXT. PRINTER OFF ?(Y/N)

This query indicates that the external printer (fed through the parallel port of the CE-158) is currently considered (by the communications package) to be off. Thus, data received via the RS-232 port will be routed to the PC-1500 for display on its LCD (or to the CE-150 if it is in use). The various option queries usually operate in a flip-flop mode. Responding (Y)es to the above query or merely pressing the ENTER key would leave the present condition unchanged. However, responding (N)o would cause the condition to switch to the opposite state (external printer considered activated). If this was done, then pressing the "Ext" softkey would result in the option now appearing as:

EXT. PRINTER ON ?(Y/N)

Note that now the printer is considered to be activated and the query is to confirm or alter this status.

Pressing the softkey under the "Trc" legend provides the opportunity to set or clear what Sharp refers to as a communications "trace" mode. When selected, this mode directs keyboard inputs to be reflected on the device connected to the parallel port. Otherwise, such inputs are shown on the LCD of the PC.

The softkey under "Dsp" is used to select whether horizontal scrolling of input lines is to occur or whether data is to essentially be displayed in maximum of 26-character blocks. This latter option is called the "clean text" mode. This mode of operation seems somewhat easier to read than a continuously scrolling line, at least when using a time-sharing service. Its selection is confirmed by the message:

CLEAN TEXT ON ?(Y/N)

Finally, in the output section, "Etx" is used to label the softkey that permits the designation of the red CL key for special uses. When this option is activated, depression of the CL key will cause the unit to transmit the ETX code (&03). Depression of SHIFT/CL will cause a 240 millisecond spacing condition to be transmitted (sometimes known as a "line break"). If the "Etx" option is not selected, then the CL key is essentially idle during communication operations. As with other options, the mode toggles between selected and non-selected as indicated by the displayed queries:

CL=ETX ON ?(Y/N)

or

CL=ETX OFF ?(Y/N)

To get out of the "output parameters" mode, you need simply press the line scroll up (or down) key. This will bring up one of the other previously mentioned menus. To review some of the other options (and thus capabilities) that can be selected, let us examine those other menus in greater detail.

The "Setup" menu has legends over three softkeys. You can make selections that enable you to establish an "automatic sign-on buffer", to define operations using the Reserve keys, to specify the length of a memory-resident communications storage area, and to customize communication parameters such as baud rate, number of data bits and number of stop bits in a character.

The "Operate" menu provides a means to select from among several operational modes such as "paged", line-at-a-time or continuous reception.

The "Protocol" menu enables one to select whether the system will automatically transmit XON and XOFF codes and how information typed on the keyboard is "echoed" on the system display device.

The communications software also has other features. There is not sufficient space to go into all the details and capabilities in this article. However, a significant capability is that of being able to "capture" information in a communications buffer. This buffer can then be scanned using "window" directives and played back on an external printer through the parallel port, if desired.

Alas, this scanning/playback operation can only take place when incoming data has been halted. Additionally, it takes a significant amount of time to search and locate a particular piece of data (or even just to get to the start of a long buffer).

I have yet to see a system that would enable one to buffer, say, 10 to 15 seconds of real time data so that pressing a switch whenever desired would permit one to obtain a "lagging" hardcopy of pertinent data. Of course, with what is now known about machine language programming on the PC-1500 and the BASIC communication statements provided in the CE-158, it might be possible for some enterprising souls to put together a truly customized communications package. There is no doubt that machine language methods would have to be used in order to provide the kind of real-time buffering capabilities necessary to implement a practical system.

It Is a Start

The CE-158 does has a lot of capability and offers unlimited potential for users to call upon the enhanced BASIC statements to tailor their own communications programs. Until such time as Sharp sees fit to produce a somewhat more readable manual, however, users should be prepared to do a lot of experimenting on their own in order to fully understand and utilize its capabilities. The material in this article should assist newcomers in getting a handle on this PC peripheral.

FROM THE WATCH POCKET

The new Radio Shack Model 100 Portable Computer is quite impressive. Putting a large display in the unit makes it a lot easier to perform word processing tasks and to follow the continuous stream of data one receives when connected to a timesharing network. Leaving out a printer, but providing several interfaces for connecting an external one seems like a sensible, cost-effective trade-off. It is a functionally attractive unit and I predict it will do well. While it is similar in size and some other respects to the Epson HX-20, it does not utilize the same kind of CPU chip. Speculation by others that it was produced by NEC seems doubtful. Radio Shack declines to identify the subcontractor.

Sharp CE-158 Needs a New Manual

After having seen the Instruction Manual for the CE-158 interface, I do not wonder that confusion abounds concerning this RS-232 device. In many places the manual is more of a hindrance than a help. Take, for example, this small quotation:

"Because this unit is a data terminal, connector signals are designed for transmission between data terminal and data communication unit. Therefore, it needs to use the cable of which signal flow is inverted, if the peripheral unit is also a data terminal because it differes (sic) in signal flow."

The sad fact is that one must first interpret the stilted English before one can begin to extract the desired technical information. With a device as complicated as the CE-158, capable of being used in a variety of arrangements, the added burden of literally having to translate each sentence makes practical application of the device quite difficult for many potential users.

That is indeed too bad, for the unit adds significant capability to a PC-1500. Not only can it give the PC the ability to communicate with a larger computer (such as via phone lines using a modem), but it can also enable it to dump listings and data to a full-sized printer. Theoretically, it can even serve as a link between the PC and a mass storage memory device such as a floppy disk. The quick loading of programs from such a peripheral is sorely needed by many PC users.

The problem is that using it with a wide variety of devices requires that one configure it in different ways. Indeed, even the wiring of the connecting cables must be altered depending upon each specific application. In other words, you have to know what you are doing. For this you need, at the very least, a comprehensible instruction manual.

Sharp had better get busy and have this manual re-written if they want to see decent sales of this device in the U.S. market. I don't think the average citizen will tolerate this kind of gibberish to describe such a sophisticated peripheral. Come on, Sharp, this could be a great product if people understood how to make it work!

Here and There

If anyone is still interested in a PC-1, Radio Shack is selling them at $99.95. Gee, is that supposed to be a bargain? The new Sharp PC-1250 with all the capabilities of the PC-1 and more has a list price of just $109.95 and can be purchased at many U.S. outlets for less than what Tandy wants for a PC-1. Matter of fact, some discounters in the New York area are trying to unload PC-1211s (the Sharp equivalent of the PC-1) complete with the optional printer/cassette interfaces for less than $100.00!

ARCsoft Publishers, PO Box 132, Woodsboro, MD 21798, has produced a new book carrying the title *Practical PC-2/PC-1500 Pocket Computer Programs*. Authored by Jim Cole, the new title is said to contain a collection of 36 "new and practical" programs for these PCs. Price is $7.95 and should soon be available in many computer-oriented bookstores. Write to the company for a current catalog as they carry a number of titles for PC-1/PC-2 and PC-1211/PC-1500 models. Many of their titles are especially good for beginners.

Robert Sutliff of New York points out that the problem with the NOT function outlined by Norlin Rober (Issue 21 of *PCN*) can be circumvented by using the form:

NOT (comparison) + 2

Thus, a statement such as IF NOT(1=2)+2 BEEP 1 will produce a beep, while IF NOT(1=1)+2 BEEP 1 will not produce a beep. He says this format will work regardless of the ROM version being used in the PC. Thanks for that tip, Robert.

Battery Lane Publications is said to be preparing a guidebook for the new Epson HX-20 Portable Computer. HX-20 users are invited to share their experiences on that machine with the guidebook editor. Product developers (both software and hardware) are invited to submit material for review consideration. Correspondence should be addressed to: HX-20 Editor, Battery Lane Information Services, 14704 Seneca Castle Court, Gaithersburg, MD 20878.

Scuttlebutt running around now says that Sharp may have 16K RAM modules that plug into the back of the PC-1500 (where 8K modules now go) available by September.

The ROM software modules for the PC-1500, which have a long history of delays, still are not available. However, there are rumors that production evaluation samples are currently being examined. A limited supply of the modules may be available in April. I will believe it when I actually see one.

I will bet it does not take long for people to crack the machine codes used in the PC-1250/1251. Although it is a different CPU than that used in the PC-1500, many *PCN* readers got plenty of "cracking" experience working on the PC-1500/PC-2. It would be nice, however, if Sharp just published the codes as an appendix in their operating manuals. It would save people a lot of work and quite possibly gain them some customers. They say they want to crack the educational market. Well, lots of Computer Science instructors would love to be able to give students a course in machine level programming while having the pupils practice on a low cost PC. The capability is there, why not tell people enough to enable them to play with it if they really want to?

A numbers of readers have inquired as to whether anyone was installing anything similar to the RAM expansion package described in Issue 23 of *PCN* on a commercial basis. I recently received word of an outfit in West Germany that apparently has a number of memory kits for the PC-1500. You might try writing for additional information to: Walter Speicel, Pfarrstr. 10 Postfach 1168, D-7320 Goppingen, West Germany. Watch out, though, the literature you receive may be entirely in German!

By the way, that new Radio Shack Model 100 contains an 80C85 CPU chip. This is believed to be a CMOS version of the popular 8085. With PEEK and POKE provided on the machine, much will be possible!

The next edition of *PCN*, Issue 25, is scheduled for publication in June. (Remember that during May!) — *Nat Wadsworth, Editor*

The POCKET COMPUTER NEWSLETTER is Available!

By Subscription Only: for a calendar year period (January – December). You get all issues published to date for the calendar year in which you subscribe, at the time you subscribe.

MC/VISA Phone Subscriptions: (203) 888-1946

☐ 1982 Regular Subscriber (Issues 11 – 20). $30.00 in U.S. (U.S. $36.00 to Canada. U.S. $45.00 elsewhere.)

☐ 1982/83 Subscriber (Issues 11 – 30). $60.00 in U.S. (U.S. $72.00 to Canada. U.S. $90.00 elsewhere.)

☐ 1983 Regular Subscriber (Issues 21 – 30). $36.00 in U.S. (U.S. $42.00 to Canada. U.S. $50.00 elsewhere.)

Orders must be accompanied by payment in full. We do not issue invoices for the POCKET COMPUTER NEWSLETTER.

Thank you for your remittance.

Name:_____

Addr: _____

City: _____ State: ___ Zip: _____

MC/VISA #: _____ Expires: _____

Signature: _____

P.O. Box 232, Seymour, CT 06483